## CLAIMS

1. In a Java computing environment, a Java Bytecode instruction suitable for execution by an inventive Java virtual machine in said Java computing environment, wherein said inventive Java Bytecode instruction operates to retrieve a string representation associated with said Java object, thereby allowing said string representation to be determined without invoking a Java method.

2. A Java Bytecode instruction as recited in claim 1, wherein said Java Bytecode instruction further operates to:

pop a reference to said Java Bytecode instruction from the top of an execution stack;

determine a string representation of a field associated with said Java object; and

push a reference to said string representation of said field on top of said execution stack.

3. A Java Bytecode instruction as recited in claim 1, wherein said Java Bytecode instruction is executed in an embedded system.

4. A Java virtual machine operating in a Java computing environment, said Java virtual machine capable of determining a string representation associated with a Java object, wherein said virtual machine determines said string representation of said Java object without invoking a Java "to_string" method.

5. Java virtual machine as recited in claim 4, wherein said Java virtual machine executes an inventive Java Bytecode instruction, said inventive Java Bytecode instruction operating to determine said string representation associated with said Java object; thereby allowing said string representation to be determined without invoking a Java method.

6.   A Java virtual machine as recited in claim 5, wherein said virtual machine operates to:

  pop a reference to said Java Bytecode instruction from the top of an execution stack;

5   determine a string representation of a field associated with said Java object; and

  push a reference to said string representation of said field on top of said execution stack.

10   7.   A Java virtual machine as recited in claim 5, wherein said Java virtual machine operates in an embedded system.

8.   In a Java computing environment, a method of retrieving a string representation for a Java object, said method comprising:

15   receiving an inventive Java Bytecode instruction in a stream of Java Bytecodes suitable for execution by a virtual machine operating in said Java computing environment, and

  wherein said inventive Java Bytecode instruction operates to determine said string representation associated with said Java object;

20   thereby allowing said string representation to be determined without invoking a Java method.

9.   A method as recited in claim 8, wherein said method further comprises:

  popping a reference to a Java object from an execution stack

25   determining a string representation of a field associated with said Java object; and

  pushing a reference to said string representation of said field on top of said execution stack.

30   10.   A method as recited in claim 7, wherein said method further comprises: pushing a reference to said Java object on said execution stack.

11.   A method as recited in claim 8, wherein said pushing of a reference to said Java object is performed by execution of a Java Aload execution.

12. A method as recited in claim 11, wherein said method is performed by a virtual machine.

5    13. A method as recited in claim 12, wherein said virtual machine is operating in an embedded system.

14. A computer readable media including computer program code for retrieving a string representation for a Java object, said computer readable
10   media comprising:

computer program code for receiving an inventive Java Bytecode instruction in a stream of Java Bytecodes suitable for execution by a virtual machine operating in said Java computing environment, and

wherein said inventive Java Bytecode instruction operates to
15   determine said string representation associated with said Java object; thereby allowing said string representation to be determined without invoking a Java method.

15. A computer readable media as recited in claim 14, wherein said
20   computer readable media further comprises:

computer program code for popping a reference to a Java object from an execution stack;

computer program code for determining a string representation of a field associated with said Java object; and
25       computer program code for pushing a reference to said string representation of said field on top of said execution stack.

16. A computer readable media as recited in claim 15, wherein said computer readable media further comprises:
30       computer program code for pushing a reference to said Java object on said execution stack.

17.  A computer readable media as recited in claim 15, wherein said computer program code for pushing said reference is performed by executing a Java Aload instruction.

5  18.  A computer readable media as recited in claim 17, wherein said computer readable media is read by a Java virtual machine.

19.  A computer readable media as recited in claim 18, wherein said virtual machine is operating in an embedded system.